1. (Cancelled)

2. (Currently Amended) The method of ~~claim 1~~claim 6, wherein the outputting step includes mapping the bits from the systematic and parity block interleavers into a symbol mapping array wherein the systematic bits are mapped into the upper rows of the array and the parity bits are mapped into the lower rows of the array.

3. (Original) The method of claim 2, wherein in the mapping step the upper rows of the array have a higher priority than the lower rows of the array, such that the systematic bits in a symbol has a higher priority than the parity bits in a symbol.

4. (Original) The method of claim 2, wherein in the mapping step the upper rows of the array have a higher priority than the lower rows of the array, such that the systematic bits in a symbol has a higher priority than the parity bits in a symbol.

5. (Currently Amended) A method of block puncturing for turbo code based incremental redundancy, the method comprising the steps of:

coding an input data stream into systematic bits and parity bits;

loading the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

selecting a predefined redundancy; and

outputting bits from the block interleavers in a row-wise manner in accordance with the selected predefined redundancy.

~~The method of claim 1,~~ further comprising the step of selecting a coding rate dependant upon available symbol memory as defined by the equation

$$N_{\mathrm{P,max\_row}} = \min\left(\left\lfloor \frac{N_{SML}}{2\times N_{CB}\times N_{col}\times N_{ARQ\_proc}} - \frac{N_{row}}{2\times N_{CB}} - \frac{N_{tail}}{2\times N_{col}\times N_{CB}} \right\rfloor, N_{row}\right)$$

where N_{p,max_row} the maximum number of rows in the parity block interleaver that can be transmitted, N_{SML} is the total number of Soft Metric Locations provisioned at the user equipment, N_{CB} is the number of code blocks, N_{col} is the number of columns in the parity block interleaver, N_{row} is the number of rows in the parity block interleaver, N_{tail} is the number of tail bits per code block, N_{ARQ_proc} is the number of ARQ

processes currently defined in the user equipment, and given that the size of the systematic block interleaver is substantially fixed.

6. (Currently Amended) A method of block puncturing for turbo code based incremental redundancy, the method comprising the steps of:

coding an input data stream into systematic bits and parity bits;

loading the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

selecting a predefined redundancy; and

outputting bits from the block interleavers in a row-wise manner in accordance with the selected predefined redundancy.

~~The method of claim 1,~~ wherein the ~~choosing~~ selecting step includes implementing a redundancy version for selecting coded bits from the block interleavers, the redundancy version includes substeps of:

setting of one or more starting rows $\alpha_{i,j}$ for each redundancy version j, where $i \in \{S, P\}$ and $\alpha_{i,j} \in \{1, \cdots, N_{i,\max\_row}\}$; and

reading the coded bits from the selected starting rows in a pre-defined order and ratio, wherein if the end of the $N_{i,\max\_row}{}^{th}$ interleaver is reached, reading shall continue from the first row of a predefined interleaver which includes one of the systematic and parity block interleavers.

7. (Currently Amended) A method of block puncturing for turbo code based incremental redundancy, the method comprising the steps of:

coding an input data stream into systematic bits and parity bits;

loading the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

selecting a predefined redundancy; and

outputting bits from the block interleavers in a row-wise manner in accordance with the selected predefined redundancy.

~~The method of claim 1,~~ wherein the ~~choosing~~ selecting step includes implementing a redundancy version for selecting coded bits from one of the group of:

setting the starting rows as the respective top rows of both the systematic and parity interleavers, and reading the coded bits of the systematic interleaver from its respective starting row to completion before the remaining coded bits are read from the parity interleaver starting at its respective starting row;

setting the starting row as $N_{p,max\_row}/4$ of the parity interleaver, and reading the coded bits from the parity interleaver starting at its starting row;

setting the starting row as $N_{p,max\_row}/2$ for the parity interleaver, and reading the coded bits from the parity interleaver starting at its starting row;

setting the starting row as $3 \times N_{p,max\_row}/4$ for the parity interleaver, and reading the coded bits from the parity interleaver starting at its starting row;

setting the starting rows as the top row of the parity interleaver and $N_{S,max\_row}/2$ for the systematic interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at its their respective starting rows;

setting the starting rows as $N_{p,max\_row}/4$ for the parity interleaver and $3 \times N_{S,max\_row}/4$ for the systematic interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at their respective starting rows;

setting the starting rows as the top row of the systematic interleaver and $N_{p,max\_row}/2$ for the parity interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at their respective starting rows; and

setting the starting rows as $3 \times N_{p,max\_row}/4$ for the parity interleaver and $N_{S,max\_row}/4$ for the systematic interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at their respective starting rows.

8. (Cancelled)

9. (Cancelled)

10. (Cancelled)

11. (Currently Amended) A method of block puncturing for turbo code based incremental redundancy, the method comprising the steps of:

turbo coding an input data stream into systematic bits and parity bits;

loading the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

selecting a predetermined redundancy; and

mapping the bits from the systematic and parity block interleavers into a symbol mapping array in a row-wise manner in accordance with the selected redundancy, wherein the systematic bits are mapped into the upper rows of the array and the parity bits are mapped into the lower rows of the array~~The method of claim 8~~, further comprising the step of selecting a coding rate dependant upon available symbol memory as defined by the equation

$$N_{\mathrm{P,max\_row}} = \min\left(\left\lfloor \frac{N_{SML}}{2\times N_{CB}\times N_{col}\times N_{ARQ\_proc}} - \frac{N_{row}}{2\times N_{CB}} - \frac{N_{tail}}{2\times N_{col}\times N_{CB}} \right\rfloor, N_{row}\right)$$

where N_{p,max_row} the maximum number of rows in the parity block interleaver that can be transmitted, N_{SML} is the total number of Soft Metric Locations provisioned at the user equipment, N_{CB} is the number of code blocks, N_{col} is the number of columns in the parity block interleaver, N_{row} is the number of rows in the parity block interleaver, N_{tail} is the number of tail bits per code block, N_{ARQ_proc} is the number of ARQ processes currently defined in the user equipment, and given that the size of the systematic block interleaver is substantially fixed.

12. (Currently Amended) A method of block puncturing for turbo code based incremental redundancy, the method comprising the steps of:

turbo coding an input data stream into systematic bits and parity bits;

loading the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

selecting a predetermined redundancy; and

mapping the bits from the systematic and parity block interleavers into a symbol mapping array in a row-wise manner in accordance with the selected redundancy, wherein the systematic bits are mapped into the upper rows of the array and the parity bits are mapped into the lower rows of the array;~~The method of claim 8,~~

wherein the ~~choosing~~ selecting step includes implementing a redundancy version for selecting coded bits from the block interleavers the redundancy version includes substeps of:

setting of one or more starting rows $\alpha_{i,j}$ for each redundancy version j, where $i \in \{S,P\}$ and $\alpha_{i,j} \in \{1,\cdots,N_{i,\max\_row}\}$; and

reading the coded bits from the selected starting rows in a pre-defined order and ratio, wherein if the end of the $N_{i,\max\_row}{}^{th}$ interleaver is reached, reading shall continue from the first row of a predefined interleaver which includes one of the systematic and parity block interleavers.

13. (Currently Amended) A method of block puncturing for turbo code based incremental redundancy, the method comprising the steps of:

turbo coding an input data stream into systematic bits and parity bits;

loading the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

selecting a predetermined redundancy; and

mapping the bits from the systematic and parity block interleavers into a symbol mapping array in a row-wise manner in accordance with the selected redundancy, wherein the systematic bits are mapped into the upper rows of the array and the parity bits are mapped into the lower rows of the array;~~The method of claim 8~~,

wherein the ~~choosing~~ selecting step includes implementing a redundancy version for selecting coded bits from the systematic and parity interleavers from one of the group of:

setting the starting rows as the respective top rows of both the systematic and parity interleavers, and reading the coded bits of the systematic interleaver from its respective starting row to completion before the remaining coded bits are read from the parity interleaver starting at its respective starting row;

setting the starting row as $N_{p,max\_row}/4$ of the parity interleaver, and reading the coded bits from the parity interleaver starting at its starting row;

setting the starting row as $N_{p,max\_row}/2$ for the parity interleaver, and reading the coded bits from the parity interleaver starting at its starting row;

setting the starting row as $3 \times N_{p,max\_row}/4$ for the parity interleaver, and reading the coded bits from the parity interleaver starting at its starting row;

setting the starting rows as the top row of the parity interleaver and $N_{S,max\_row}/2$ for the systematic interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at its their respective starting rows;

setting the starting rows as $N_{p,max\_row}/4$ for the parity interleaver and $3 \times N_{S,max\_row}/4$ for the systematic interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at their respective starting rows;

setting the starting rows as the top row of the systematic interleaver and $N_{p,max\_row}/2$ for the parity interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at their respective starting rows; and

setting the starting rows as $3 \times N_{p,max\_row}/4$ for the parity interleaver and $N_{S,max\_row}/4$ for the systematic interleaver, and reading the coded bits equally from the systematic and parity interleaver starting at their respective starting rows.

14. (Cancelled)

15. (Cancelled)

16. (Cancelled)

17. (Cancelled)

18. (Currently Amended) A turbo coder with block puncturing for incremental redundancy, comprising:

a channel coder operable to code an input data stream into systematic bits and parity bits;

a first interleaver coupled to the channel coder, the first interleaver operable to load the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

a redundancy version selector coupled to the first interleaver, the redundancy version selector operable to select a predefined redundancy;

a bit priority mapper coupled to the redundancy version selector, the bit priority mapper operable to map bits from the block interleavers in a row-wise manner in accordance with the selected predefined redundancy~~The coder of claim 14~~, further

comprising a symbol memory, and wherein the redundancy version selector also selects a. coding rate dependant upon available symbol memory as defined by the equation

$$N_{\mathrm{P,max\_row}} = \min\left(\left\lfloor \frac{N_{SML}}{2\times N_{CB}\times N_{col}\times N_{ARQ\_proc}} - \frac{N_{row}}{2\times N_{CB}} - \frac{N_{tail}}{2\times N_{col}\times N_{CB}} \right\rfloor, N_{row}\right)$$

where N_{p,max_row} the maximum number of rows in the parity block interleaver that can be transmitted, N_{SML} is the total number of Soft Metric Locations provisioned at the user equipment, N_{CB} is the number of code blocks, N_{col} is the number of columns in the parity block interleaver, N_{row} is the number of rows in the parity block interleaver, N_{tail} is the number of tail bits per code block, N_{ARQ_proc} is the number of ARQ processes currently defined in the user equipment, and given that the size of the systematic block interleaver is substantially fixed.

19. (Currently Amended) A turbo coder with block puncturing for incremental redundancy, comprising:

a channel coder operable to code an input data stream into systematic bits and parity bits;

a first interleaver coupled to the channel coder, the first interleaver operable to load the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

a redundancy version selector coupled to the first interleaver, the redundancy version selector operable to select a predefined redundancy;

a bit priority mapper coupled to the redundancy version selector, the bit priority mapper operable to map bits from the block interleavers in a row-wise manner in accordance with the selected predefined redundancy~~The coder of claim 14~~, wherein the predefined redundancy comprises a redundancy version for the coded bits of the block interleavers wherein the redundancy version sets one or more starting rows $\alpha_{i,j}$ for each redundancy version j, where $i \in \{S, P\}$ and $\alpha_{i,j} \in \{1, \cdots, N_{i,\mathrm{max\_row}}\}$ and reads the coded bits from the selected starting rows in a pre-defined order and ratio, wherein if the end of the $N_{i,\mathrm{max\_row}}$ th interleaver is reached, reading shall continue

from the first row of a predefined interleaver which includes one of the systematic and parity block interleavers.

20. (Currently Amended) A turbo coder with block puncturing for incremental redundancy, comprising:

a channel coder operable to code an input data stream into systematic bits and parity bits;

a first interleaver coupled to the channel coder, the first interleaver operable to load the systematic bits and parity bits into respective systematic and parity block interleavers in a column-wise manner;

a redundancy version selector coupled to the first interleaver, the redundancy version selector operable to select a predefined redundancy;

a bit priority mapper coupled to the redundancy version selector, the bit priority mapper operable to map bits from the block interleavers in a row-wise manner in accordance with the selected predefined redundancy~~The coder of claim 14~~, wherein the predefined redundancy comprises a redundancy version for the coded bits of the systematic and parity interleavers selected from one of the group of:

set the starting rows as the respective top rows of both the systematic and parity interleavers, wherein the coded bits of the systematic interleaver are read from its respective starting row to completion before the remaining coded bits are read from the parity interleaver starting at its respective starting row;

set the starting row as $N_{p,max\_row}/4$ of the parity interleaver, wherein the coded bits are read from the parity interleaver starting at its starting row;

set the starting row as $N_{p,max\_row}/2$ for the parity interleaver, wherein the coded bits are read from the parity interleaver starting at its starting row;

set the starting row as $3 \times N_{p,max\_row}/4$ for the parity interleaver, wherein the coded bits are read from the parity interleaver starting at its starting row;

set the starting rows as the top row of the parity interleaver and $N_{S,max\_row}/2$ for the systematic interleaver, wherein the coded bits are read equally from the systematic and parity interleaver starting at its their respective starting rows;

set the starting rows as $N_{p,max\_row}/4$ for the parity interleaver and $3 \times N_{S,max\_row}/4$ for the systematic interleaver, wherein the coded bits are read equally from the systematic and parity interleaver starting at their respective starting rows;

set the starting rows as the top row of the systematic interleaver and $N_{p,max\_row}/2$ for the parity interleaver, wherein the coded bits are read equally from the systematic and parity interleaver starting at their respective starting rows; and

set the starting rows as $3 \times N_{p,max\_row}/4$ for the parity interleaver and $N_{S,max\_row}/4$ for the systematic interleaver, wherein the coded bits are read equally from the systematic and parity interleaver starting at their respective starting rows.

21. (Newly Presented) The method of claim 7, wherein the outputting step includes mapping the bits from the systematic and parity block interleavers into a symbol mapping array wherein the systematic bits are mapped into the upper rows of the array and the parity bits are mapped into the lower rows of the array.

22. (Newly Presented) The method of claim 21, wherein in the mapping step the upper rows of the array have a higher priority than the lower rows of the array, such that the systematic bits in a symbol has a higher priority than the parity bits in a symbol.

23. (Newly Presented) The method of claim 21, wherein in the mapping step the upper rows of the array have a higher priority than the lower rows of the array, such that the systematic bits in a symbol has a higher priority than the parity bits in a symbol.

24. (Newly Presented) The method of claim 5, wherein the outputting step includes mapping the bits from the systematic and parity block interleavers into a symbol mapping array wherein the systematic bits are mapped into the upper rows of the array and the parity bits are mapped into the lower rows of the array.

25. (Newly Presented) The method of claim 24, wherein in the mapping step the upper rows of the array have a higher priority than the lower rows of the array, such that the systematic bits in a symbol has a higher priority than the parity bits in a symbol.

26. (Newly Presented) The method of claim 24, wherein in the mapping step the upper rows of the array have a higher priority than the lower rows of the array, such that the systematic bits in a symbol has a higher priority than the parity bits in a symbol.